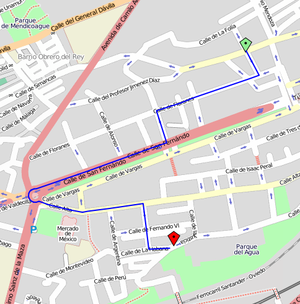
**Algoritmos de Enrutamiento**

[](http://es.wikipedia.org/wiki/Archivo:OpenStreetMap_routing_service.png)

**Enrutamiento**

Es la función de buscar un camino entre todos los posibles en una red de paquetes cuyas topologías poseen una gran conectividad. Dado que se trata de encontrar la mejor ruta posible, lo primero será definir qué se entiende por mejor ruta y en consecuencia cuál es la métrica que se debe utilizar para medirla.

**Los Algoritmos de encaminamiento pueden agruparse en:**

**Determinanticos o estáticos**

No tienen en cuenta el estado de la subred al tomar las decisiones de encaminamiento. Las tablas de encaminamiento de los nodos se configuran de forma manual y permanecen inalterables hasta que no se vuelve a actuar sobre ellas. Por tanto, la adaptación en tiempo real a los cambios de las condiciones de la red es nula. El cálculo de la ruta óptima es también off-line por lo que no importa ni la complejidad del algoritmo ni el tiempo requerido para su convergencia. Ej: algoritmo de Dijkstra. Estos algoritmos son rígidos, rápidos y de diseño simple, sin embargo son los que peores decisiones toman en general.

**Adaptativos o dinámicos**

Pueden hacer más tolerantes a cambios en la subred tales como variaciones en el tráfico, incremento del retardo o fallas en la topología. El encaminamiento dinámico o adaptativo se puede clasificar a su vez en tres categorías, dependiendo de donde se tomen las decisiones y del origen de la información intercambiada:

* Adaptativo centralizado. Todos los nodos de la red son iguales excepto un nodo central que es quien recoge la información de control y los datos de los demás nodos para calcular con ellos la tabla de encaminamiento. Este método tiene el inconveniente de que consume abundantes recursos de la propia red.
* Adaptativo distribuido. Este tipo de encaminamiento se caracteriza porque el algoritmo correspondiente se ejecuta por igual en todos los nodos de la subred. Cada nodo recalcula continuamente la tabla de encaminamiento a partir de dicha información y de la que contiene en su propia base de datos. A este tipo pertenecen dos de los más utilizados en Internet que son los algoritmos por [vector de distancias](http://es.wikipedia.org/wiki/Vector_de_distancias) y los de [estado de enlace](http://es.wikipedia.org/wiki/Estado_de_enlace).
* Adaptativo aislado. Se caracterizan por la sencillez del método que utilizan para adaptarse al estado cambiante de la red. Su respuesta a los cambios de tráfico o de topología se obtiene a partir de la información propia y local de cada nodo. Un caso típico es el encaminamiento “por inundación” cuyo mecanismo consiste en reenviar cada paquete recibido con destino a otros nodos, por todos los enlaces excepto por el que llegó.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Algoritmos por “Vector de Distancias”**  Estos métodos utilizan el algoritmo de Bellman-Ford. Busca la ruta de menor coste por el método de búsqueda indirecta El [vector de distancias](http://es.wikipedia.org/wiki/Vector_de_distancias) asociado al nodo de una red, es un paquete de control que contiene la distancia a los nodos de la red conocidos hasta el momento.Cada nodo envía a sus vecinos las distancias que conoce a través de este paquete. Los nodos vecinos examinan esta información y la comparan con la que ya tienen, actualizando su tabla de encaminamiento .Ejemplos de protocolos por vector de distancias: [RIP](http://es.wikipedia.org/wiki/RIP_(protocolo)) (versión 1 y 2), [IGRP](http://es.wikipedia.org/wiki/IGRP).ç  Se trata de uno de los más importantes junto con el de estado de enlace. Utiliza el algoritmo de Bellman-Ford para calcular las rutas. Fue el algoritmo original de ARPANET. Se usó en DECNET, IPX y Appletalk. Lo usa el protocolo [RIP](http://es.wikipedia.org/wiki/Protocolo_de_encaminamiento_de_informaci%C3%B3n) (Routing Information Protocol), que hasta 1988 era el único utilizado en Internet. También se utiliza en los protocolos propietarios ampliamente extendidos IGRP y [EIGRP](http://es.wikipedia.org/wiki/EIGRP) de [Cisco](http://es.wikipedia.org/wiki/Cisco).  **Funcionamiento**  El encaminamiento de un protocolo basado en vector de distancias requiere que un router informe a sus vecinos de los cambios en la topología periódicamente y en algunos casos cuando se detecta un cambio en la topología de la red. Comparado a los protocolos de estado de enlace, que necesitan que un router informe a todos los nodos de una red acerca de los cambios en su topología, los algoritmos de vector de distancias tienen mucho menos complejidad computacional. Además, las principales características de los diferentes algoritmos VD (vector de distancias) son siempre las mismas. El algoritmo VD se basa en calcular la dirección y la distancia hasta cualquier enlace en la red. El costo de alcanzar un destino se lleva a cabo usando cálculos matemáticos como la métrica del camino. [RIP](http://es.wikipedia.org/wiki/RIP) cuenta los saltos efectuados hasta llegar al destino mientras que [IGRP](http://es.wikipedia.org/wiki/IGRP) utiliza otra información como el retardo y el ancho de banda.  Los cambios son detectados periódicamente ya que la tabla de encaminamiento de cada router se envía a todos los vecinos que usan en mismo protocolo. Una vez que el router tiene toda la información, actualiza su propia tabla reflejando los cambios y luego informa a sus vecinos de los mismos. Este proceso se conoce también como “encaminamiento por rumor” ya que los nodos utilizan la información de sus vecinos y no pueden comprobar a ciencia cierta si ésta es verdadera o no.  El algoritmo de Bellman-Ford se adapta perfectamente al modo de aprendizaje de los nodos que “nacen”, es decir, cuando se conectan a la red. A medida que el algoritmo progresa, el nuevo nodo va adquiriendo más información sobre el resto de nodos de la red. Este algoritmo converge rápidamente cuando se conectan nuevos nodos. Por ello se suele decir que las buenas noticias viajan rápido por la red.  **Sus Limitaciones**  Un problema es el de la transmisión de malas noticias por la red tales como la ruptura de un enlace o la desaparición de un nodo. Este algoritmo converge lentamente en estos casos. Aunque el principal inconveniente de este algoritmo es el de la cuenta a infinito.  El algoritmo Bellman-Ford utilizado en VD no previene de la aparición de bucles. Aunque protocolos como IGRP están modificados para detectar bucles en la red. El problema de la cuenta a infinito es que hace que los costes o distancias se incrementen indefinidamente sin que el algoritmo llegue a converger nunca.  Para ilustrarlo, tomemos como ejemplo el de la figura.  [VectorDistancia.JPG](http://es.wikipedia.org/wiki/Archivo:VectorDistancia.JPG)   * Inicialmente A está desactivado. Cuando A se activa, B se entera de que A existe al recibir su vector distancia y actualizar su tabla indicando que A dista 1. * El nodo C se entera de que A existe porque B le indica que tiene un enlace hacia A de coste 1. Entonces C actualiza su tabla registrando una trayectoria hacia A de coste 2. * Si el nodo A se desconecta entonces B no recibe el VD de A. Sin embargo el nodo C le dice que tiene una trayectoria hasta A de distancia 2. B no sabe que la trayectoria de C a A pasa por el mismo y por tanto cree que puede llegar a A a través de C por lo que actualiza su tabla registrando la distancia 2 + 1 = 3 hasta A * En el siguiente intercambio, el nodo C comprueba que sus vecinos B y D tienen una trayectoria hasta A de distancia 3. C calcula su propia distancia hasta A en 3 + 1 = 4. En los siguientes intercambios, los nodos elevan ilimitadamente su distancia a A (cuenta a infinito).   Mientras no se interrumpa la cuenta a infinito, el algoritmo no converge. Aunque se han propuesto diversas soluciones a este problema.  **Ejemplo**  En esta red, tenemos 4 routers A, B, C,y D:[Ejemplo de vectores de distancia](http://es.wikipedia.org/wiki/Archivo:Networkabcd.svg)  Empezamos calculando las matrices de distancias para cada router. El “camino más corto” está marcado con el color verde, un “camino más corto” nuevo está indicado en amarillo   |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | | T=0 | |  |  |  |  |  | | --- | --- | --- | --- | --- | | **desde A** | **via A** | **via B** | **via C** | **via D** | | **a A** |  |  |  |  | | **a B** |  | 3 |  |  | | **a C** |  |  | 23 |  | | **a D** |  |  |  |  | | |  |  |  |  |  | | --- | --- | --- | --- | --- | | **desde B** | **via A** | **via B** | **via C** | **via D** | | **a A** | 3 |  |  |  | | **a B** |  |  |  |  | | **a C** |  |  | 2 |  | | **a D** |  |  |  |  | | |  |  |  |  |  | | --- | --- | --- | --- | --- | | **desde C** | **via A** | **via B** | **via C** | **via D** | | **a A** | 23 |  |  |  | | **a B** |  | 2 |  |  | | **a C** |  |  |  |  | | **a D** |  |  |  | 5 | | |  |  |  |  |  | | --- | --- | --- | --- | --- | | **desde D** | **via A** | **via B** | **via C** | **via D** | | **a A** |  |  |  |  | | **a B** |  |  |  |  | | **a C** |  |  | 5 |  | | **a D** |  |  |  |  | | | T=1 | |  |  |  |  |  | | --- | --- | --- | --- | --- | | **desde A** | **via A** | **via B** | **via C** | **via D** | | **a A** |  |  |  |  | | **a B** |  | 3 | 25 |  | | **a C** |  | 5 | 23 |  | | **a D** |  |  | 28 |  | | |  |  |  |  |  | | --- | --- | --- | --- | --- | | **desde B** | **via A** | **via B** | **via C** | **via D** | | **a A** | 3 |  | 25 |  | | **a B** |  |  |  |  | | **a C** | 26 |  | 2 |  | | **a D** |  |  | 7 |  | | |  |  |  |  |  | | --- | --- | --- | --- | --- | | **desde C** | **via A** | **via B** | **via C** | **via D** | | **a A** | 23 | 5 |  |  | | **a B** | 26 | 2 |  |  | | **a C** |  |  |  |  | | **a D** |  |  |  | 5 | | |  |  |  |  |  | | --- | --- | --- | --- | --- | | **desde D** | **via A** | **via B** | **via C** | **via D** | | **a A** |  |  | 28 |  | | **a B** |  |  | 7 |  | | **a C** |  |  | 5 |  | | **a D** |  |  |  |  | | | T=2 | |  |  |  |  |  | | --- | --- | --- | --- | --- | | **desde A** | **via A** | **via B** | **via C** | **via D** | | **a A** |  |  |  |  | | **a B** |  | 3 | 25 |  | | **a C** |  | 5 | 23 |  | | **a D** |  | 10 | 28 |  | | |  |  |  |  |  | | --- | --- | --- | --- | --- | | **desde B** | **via A** | **via B** | **via C** | **via D** | | **a A** | 3 |  | 25 |  | | **a B** |  |  |  |  | | **a C** | 26 |  | 2 |  | | **a D** | 31 |  | 7 |  | | |  |  |  |  |  | | --- | --- | --- | --- | --- | | **desde C** | **via A** | **via B** | **via C** | **via D** | | **a A** | 23 | 5 |  | 33 | | **a B** | 26 | 2 |  | 12 | | **a C** |  |  |  |  | | **a D** | 51 | 9 |  | 5 | | |  |  |  |  |  | | --- | --- | --- | --- | --- | | **desde D** | **via A** | **via B** | **via C** | **via D** | | **a A** |  |  | 10 |  | | **a B** |  |  | 7 |  | | **a C** |  |  | 5 |  | | **a D** |  |  |  |  | | | T=3 | |  |  |  |  |  | | --- | --- | --- | --- | --- | | **desde A** | **via A** | **via B** | **via C** | **via D** | | **a A** |  |  |  |  | | **a B** |  | 3 | 25 |  | | **a C** |  | 5 | 23 |  | | **a D** |  | 10 | 28 |  | | |  |  |  |  |  | | --- | --- | --- | --- | --- | | **desde B** | **via A** | **via B** | **via C** | **via D** | | **a A** | 3 |  | 7 |  | | **a B** |  |  |  |  | | **a C** | 8 |  | 2 |  | | **a D** | 13 |  | 7 |  | | |  |  |  |  |  | | --- | --- | --- | --- | --- | | **desde C** | **via A** | **via B** | **via C** | **via D** | | **a A** | 23 | 5 |  | 15 | | **a B** | 26 | 2 |  | 12 | | **a C** |  |  |  |  | | **a D** | 33 | 9 |  | 5 | | |  |  |  |  |  | | --- | --- | --- | --- | --- | | **desde D** | **via A** | **via B** | **via C** | **via D** | | **a A** |  |  | 10 |  | | **a B** |  |  | 7 |  | | **a C** |  |  | 5 |  | | **a D** |  |  |  |  | |   **Algoritmos de “Estado de Enlace”**  Este tipo de encaminamiento se basa en que cada nodo llegue a conocer la topología de la red y los costes (retardos) asociados a los enlaces, para que a partir de estos datos, pueda obtener el árbol y la tabla de encaminamiento tras aplicar el algoritmo de coste mínimo ([algoritmo de Dijkstra](http://es.wikipedia.org/wiki/Algoritmo_de_Dijkstra)) al grafo de la red |

Estado de enlace Se basa en que un router o encaminador comunica a los restantes nodos de la red cuáles son sus vecinos y a qué distancias está de ellos. Con la información que un nodo de la red recibe de todos los demás, puede construir un "mapa" de la red y sobre él calcular los caminos óptimos. El encaminamiento por estado de enlace nace en 1979 cuando en ARPANET sustituyó al método de vector de distancias.

**Funcionamiento**

Lo podemos dividir en cinco pasos fundamentales:

1. Descubrir a sus vecinos y sus direcciones  
2. Medir el costo a cada uno de sus vecinos  
3. Construir el paquete con la información recabada  
4. Enviar este paquete al resto de routers.  
5. Calcular la ruta mínima al resto de routers

**Determinar los vecinos de cada nodo**. Lo primero que debe hacer un router al activarse es averiguar quienes son sus vecinos. Para ello, manda un paquete especial HELLO por cada línea punto a punto. Todo router que reciba este paquete debe responder indicando su identidad.

**Cálculo del coste a los vecinos**. Para medir el retardo a cada nodo, el router manda un paquete especial ECHO a través de la línea el cual debe volver a su origen. El tiempo de ida y vuelta dividido entre dos nos da una aproximación razonable del costo a cada vecino de la red

**Elaboración de paquete de estado de enlace**. El siguiente paso consiste en que cada router construye un paquete con todos los datos que informan del estado de la red. La estructura de este paquete es la siguiente:

* Identidad del router
* Secuencia
* Edad
* Lista de nodos vecinos

El problema de esta etapa es el momento de la creación estos paquetes. Hay varias alternativas como hacerlo de manera periódica o bien cuando haya ocurrido un evento en la red como la caída de un nodo.

**Distribución del paquete de estado de enlace**. Es la parte más complicada del algoritmo. Básicamente lo que hace, es repartir el paquete por toda la red por inundación. Para controlarla, cada paquete incluye un número de secuencia que aumenta con cada paquete nuevo enviado. Cada router contiene una tabla con toda la información de tal manera que:

* Si recibe un paquete nuevo, este se envía por todas las líneas excepto por la que llega.
* Si se trata de un duplicado, lo elimina.
* Si es un paquete con secuencia menor que el mayor visto hasta el momento, lo rechaza.

A pesar de todo, surgen ciertos problemas como el reinicio de la secuencia. Si ocurre esto, se producirá un caos en la red. Este problema se soluciona usando secuencias de 32 bits, lo suficientemente grandes para no tener que poner la secuencia a 0 suponiendo que se envía un paquete por segundo. Otros conflictos surgen en el caso de caída de un router (reinicio del número de secuencia) o si se recibe un número de secuencia equivocado por haberse modificado alguno de sus bits durante la transmisión.

La solución para esto, es introducir la edad de cada paquete e ir disminuyéndola en un intervalo pequeño de tiempo. Cuando la edad llegue a 0, estos paquetes son descartados. Además, este método permite que los paquetes no circulen de manera indefinida por la red.

**Cálculo de ruta mínima**. Una vez que el router ha completado la recopilación de información, puede construir el grafo de la subred. De esta manera, se puede utilizar el algoritmo de Dijkstra para calcular el camino más corto a todos los nodos.

**Estado de Enlace vs Vector Distancia**

A continuación se realiza una comparativa entre estos dos algoritmos:

**Ancho de banda**. Puesto que la métrica de retardo es la longitud de la cola, el vector distancia no considera el ancho de banda usado. Antes de 1979 el máximo ancho de banda era de 56Kb posteriormente se modernizaron las líneas a 230Kbps o incluso a 1,5Mbps lo que hizo necesario el uso de mejores técnicas.

**Convergencia**. El algoritmo por vector distancia tarda demasiado en converger aún con la técnica del horizonte dividido.

**Información de la red**. En encaminamiento por vector distancia, cada router envía información sólo a sus vecinos, pero esta es sobre toda la red. Sin embargo el encaminamiento por EE envía a todos los nodos de la red, pero su información es relativa a sus vecinos. Además el enrutamiento por vector distancia no permite conocer la topología de la red.

**Capacidad y uso de memoria**. Con algoritmos basados en estado de enlace, el tráfico de la red siempre es el mismo sin depender del tamaño de la red. Con vectores distancia, se transmiten vectores de un tamaño proporcional al número de nodos. El routing por vector distancia sólo guarda las distancias al resto de nodos. Con estado de enlace se ha de almacenar además la topología de la red.

**Sucesos en la red**. Al no tener información sobre la topología, el routing por vector distancia no se adapta tan bien a los cambios en la red como el basado en estado de enlace. Sin embargo, el encaminamiento basado en vector distancia es mucho más sencillo que el de estado de enlace, lo que en ocasiones puede resultar bastante útil.